

Séance 3 : Structure conditionnelle (if...else...) : Si... Alors... Sinon...

Définition : Un script permet d'exécuter une liste d'instructions les unes à la suite des autres, mais un script peut aussi n'exécuter certaines instructions que si une ou plusieurs conditions sont remplies.

En Python : On a : **if** *condition* :
 instructions 1
 else :
 instructions 2

Quand il n'y a pas d'instructions à exécuter dans le cas où la condition est fausse, on écrit juste :
if *condition* :
 instructions 1

Le mot clé **if** introduit la boucle. Les deux points **:** à la fin de la ligne ouvrent le bloc indenté (décalé vers la droite) comportant les instructions à effectuer (*instructions 1*). C'est le retour vers la gauche, à la ligne suivante, qui marque la sortie de cette partie d'instructions.

Remarque : il peut être lourd d'imbriquer des structures conditionnelles les unes dans les autres. Python dispose de **elif** (contraction de else if) pour ce type de cas.

Symboles utilisés pour les conditions :

<i>maths</i>	<	>	≤	≥	=	≠
<i>Python</i>	<	>	<=	>=	==	!=

Quelques opérations :

Dans la division euclidienne de a par b : • $a // b$: quotient (entier) • $a \% b$: reste
 Élever a à la puissance n : • $a ** n$
 Calculer 4×10^3 : • $4e3$

Exemple : On calcule le reste de la division par 2 d'un entier. Il n'y a que deux cas : si le reste est 0, cet entier est pair et, sinon, il est impair. On crée alors un fichier, *parite*, pour y écrire le script :

```

parite.py      004/008
print("parite(n)")
def parite(n):
    if n%2 ==0:
        reponse="pair"
    else:
        reponse="impair"
    return(reponse)
    
```

→ **Coup de pouce vidéo** : <https://www.youtube.com/watch?v=AO0x9rqJ7uA>
 et rappels dans le livre page 345

Application : Soit f une fonction polynôme de degré 2, telle que $f(x) = ax^2 + bx + c$ avec $a \neq 0$.
 Écrire un script, *seconddeg*, sur ta calculatrice utilisant une fonction et qui, connaissant les valeurs de a , b et c , donne la valeur du discriminant et les solutions de l'équation $f(x) = 0$.
N. B. : il faudra importer la bibliothèque « math » pour pouvoir utiliser racine carrée $\text{sqrt}(x)$.

```
secondeg.py 001/016 ▶
print("solEqua2deg(a,b,c)")
from math import *
def solEqua2deg(a,b,c):
    d=b**2-4*a*c
    print("Delta=b*b-4*a*c=",d)
    if d<0:
        print("Pas de solution")
    if d==0:
        print("Une solution")
        print("x0 =",-b/(2*a))
    if d>0:
        print("Deux solutions")
        print("x1 =",(-b+sqrt(d))/(2*a))
        print("x2 =",(-b-sqrt(d))/(2*a))
    return
```

FILE RUN SYMBOL CHAR A↔a ▶